AJAX – Asynchronous JavaScript And XML

Dr. Shaukat Ali

Department of Computer Science

University of Peshawar

Overview

- What is AJAX ?

- Why AJAX ?

- AJAX Vs Classical Web Approach.

- How AJAX Works ?

- XMLHttpRequest ?

- How to make an AJAX application ?

- Live AJAX Examples.

Back Then

And You Get This!

These Days …

AJAX

The magic spell behind this is :

AJAX

What is AJAX ?

- A programming language --- no.
- Ajax is a new technology --- not exactly

"It is a methodology on using several web technologies together, in an effort to close the gap between the usability and interactivity of a desktop application and the ever demanding web application"

What is AJAX? (cont)

- It is a technique used to develop interactive web applications that are able to process a user request immediately.

- Ajax can selectively modify a part of a page displayed by the browser, and update it without the need to reload the whole document with all images, menus, etc.

What is AJAX ?(cont)

- The word "**Asynchronous**" in AJAX means that the request to the server will be made. The response will be made available by the server after it has finished processing the request, without having to wait for it explicitly, to come back. i.e. you *don't have to wait for an answer.*

Order of Developments

- Internet Explorer introduces the concept of IFrame element in 1996.(a technique that helps in loading the contents of a web page.)

- In the year 1998, Microsoft introduces another technique, called "Microsoft's Remote Scripting" as a replacement to the older techniques.

Order of Developments

- A year later, in 1999, Microsoft introduces the XMLHttpRequest object, an ActiveX control, in IE 5.

- The term AJAX is coined on February 18, 2005, by **Jesse James Garret in a short** essay published a few days after Google released its Maps application.

- Finally, in the year 2006, the W3C (World Wide Web Consortium) announces the release of the first draft which includes the specification for the object (XHR) and makes it an official web standard.

Why AJAX?

- AJAX enables a much *better user experience for Web sites and* applications.

- Developers can now provide user interfaces that are nearly as responsive and rich as more traditional Windows Forms applications.

- These benefits have been shown to dramatically reduce software maintenance costs and increase its reach. You can use AJAX to load specific portions of a page that need to be changed.

- It further reduces network traffic.

Why AJAX? (cont)

- The common problem is WebPages always "RELOAD" and never get "UPDATED", creating a break in user interaction.

- Users wait for the entire page to load even if a single piece of data is needed

- Restriction on single Request/Response.

- Need for complicated UI components.

Technologies Involved

- **JavaScript** – for binding everything together.

- **HTML & Cascading Style Sheets** (CSS), for presentation and to provide style classes for the component styling.

- **XMLHttpRequest** – provides asynchronous data retrieval.

- **Document Object Model** (DOM), for dynamic display and interaction and provides a tree structure for the components placed in the web page.

- **Extensible Markup Language** (XML), Format to send data from client to server, though other formats like JSON may also be used.

How it Works

Using JavaScript, an instance of the xmlHttpRequest object is created. The HttpRequest is then sent.

The client processes the returned XML document using JavaScript and updates the page content.

Internet

The HttpRequest is processed by the server. A response is created and returned as XML data to the client.

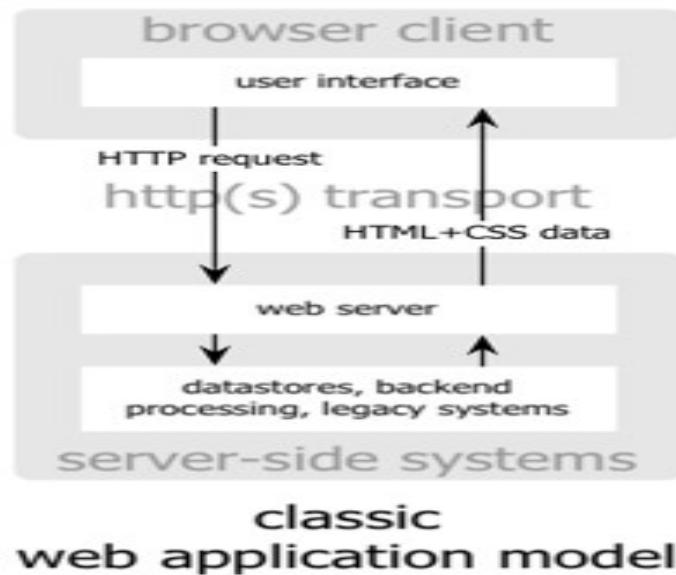# AJAX approach Vs Classical approach

Diagram Reference : www.adaptivepath.com

AJAX approach Vs Classical approach (Cont)

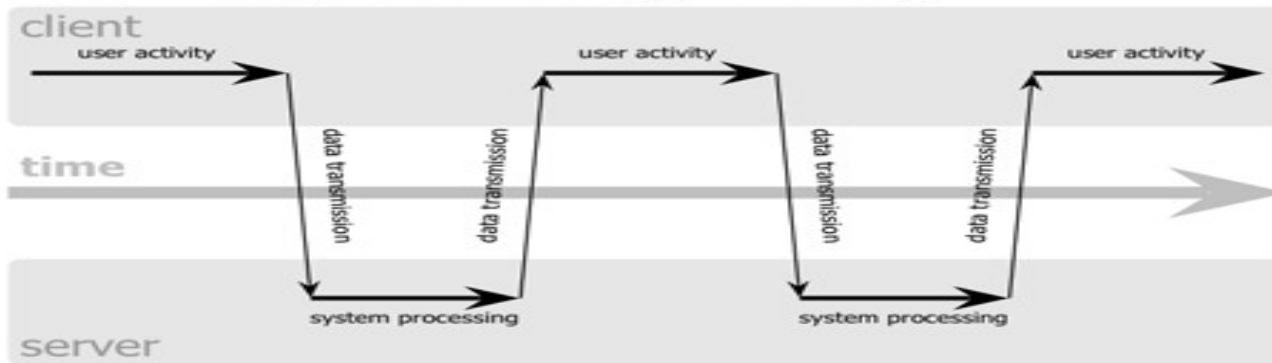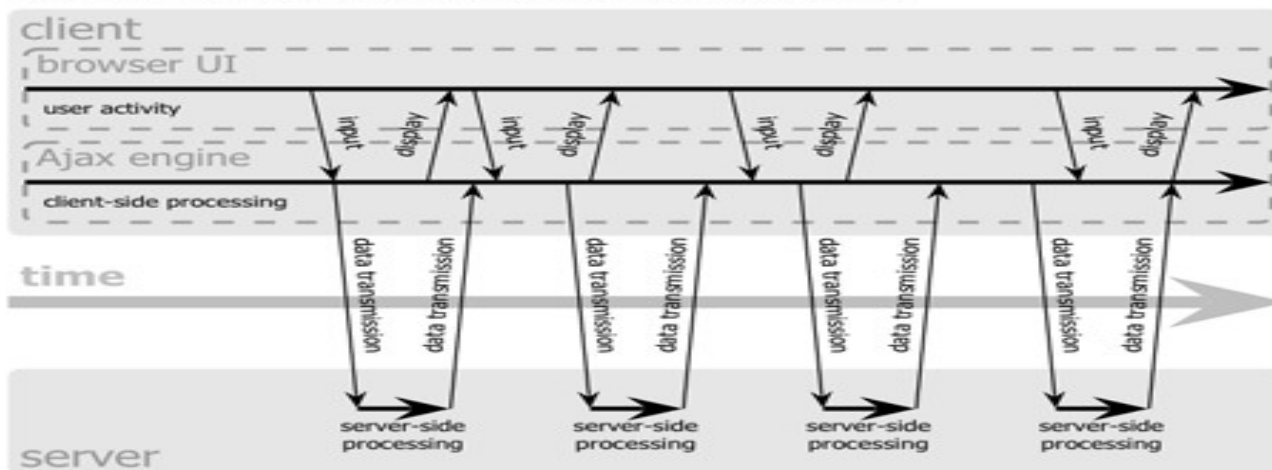classic web application model (synchronous)

Ajax web application model (asynchronous)

Diagram reference: www.adaptivepath.com

How AJAX Works ?

- A JavaScript function creates and configures an XMLHttpRequest object on the client, and specifies a JavaScript callback function.

- The XMLHttpRequest object makes an asynchronous call to the web server.

- The web server processes the request and returns an XML document that contains the result.

- The XMLHttpRequest object calls the callback function and exposes the response from the web server so that the request can be processed.

- The client updates the HTML DOM representing the page with the new data.

How AJAX Works ? (cont)
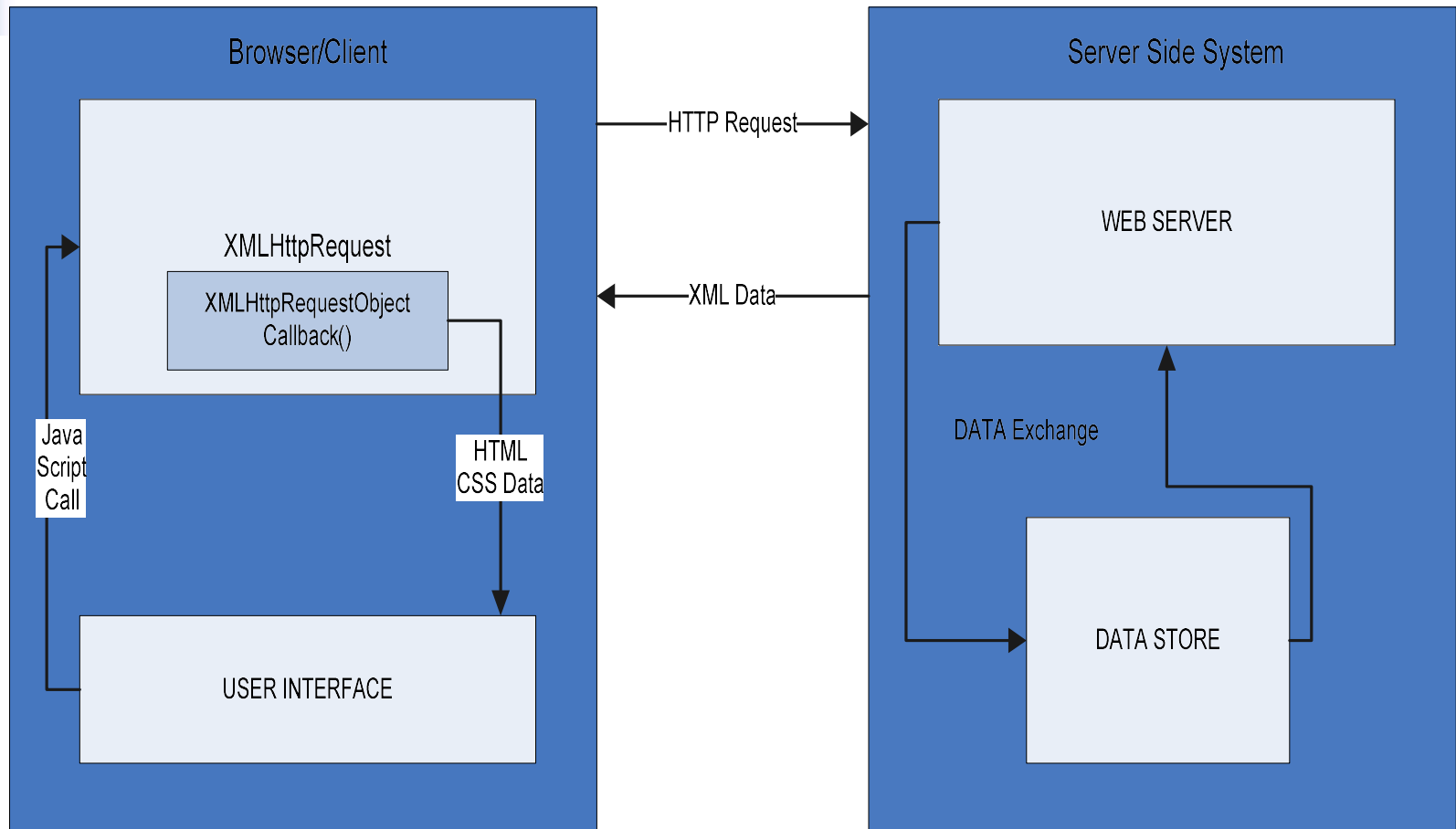
XMLHttpRequest ?

- XMLHttpRequest object is the key to Ajax programming.

- It's main purpose is to put an asynchronous http request to the web server.

- Because of this asynchronous call to the web server, you are allowed to continue using the page without the interruption of a browser refresh and the loading of a new or revised page.

- This object has few properties.

XHR Object

Table 3-1 XMLHttpRequest Object Properties for Internet Explorer

Property	Means	Read/write
onreadystatechange	Holds the name of the event handler that should be called when the value of the readyState property changes	Read/write
readyState	Holds the state of the request	Read-only
responseBody	Holds a response body, which is one way HTTP responses can be returned	Read-only
responseStream	Holds a response stream, a binary stream to the server	Read-only
responseText	Holds the response body as a string	Read-only
responseXML	Holds the response body as XML	Read-only
status	Holds the HTTP status code returned by a request	Read-only
statusText	Holds the HTTP response status text	Read-only

XHR Object

Table 3-2	XMLHttpRequest Object Methods for Internet Explorer
Method	**Means**
abort	Aborts the HTTP request
getAllResponseHeaders	Gets all the HTTP headers
getResponseHeader	Gets the value of an HTTP header
open	Opens a request to the server
send	Sends an HTTP request to the server
setRequestHeader	Sets the name and value of an HTTP header

Properties of XMLHttpRequest

- Property 1: objXML*Http.onreadystatechange*

    This property holds the reference of function which is going to process the response from the server.

objXMLHttp.onreadystatechange = procRequest;

* "procRequest " is the function which will process the response

Properties of XMLHttpRequest

- Property 2 : objXML*Http. readyState*

   This property holds the status of server response.

objXMLHttp.readyState = [state];

State	Description
0	The request is not initialized
1	The request has been set up
2	The request has been sent
3	The request is in process
4	The request is complete

Properties of XMLHttpRequest

- Property 3: objXML*Http.responseText*

    This property retrieves the data sent back from server.

    var objVal = objXMLHttp. responseText;

    While the responseText is used to return text ,

    responseXML can be used to return an XML document object.

    var   xmlDoc;

    xmlDoc = objXMLHttp.responseXML.documentElement

How to make an Ajax application ?

```html
<html>
 <head>
   <title>Ajax at work</title>
   <script language = "javascript">
     var XMLHttpRequestObject = false;
     if (window.XMLHttpRequest) {
       XMLHttpRequestObject = new XMLHttpRequest(); //for mozilla
     } else if (window.ActiveXObject) {
XMLHttpRequestObject = new ActiveXObject("Microsoft.XMLHTTP"); // IE
     }
     function getData(dataSource, divID)
     {
       if(XMLHttpRequestObject) {
         var obj = document.getElementById(divID);
         XMLHttpRequestObject.open("GET", dataSource);
         XMLHttpRequestObject.onreadystatechange = function()
         {
           if (XMLHttpRequestObject.readyState == 4 &&
             XMLHttpRequestObject.status == 200) {
               obj.innerHTML = XMLHttpRequestObject.responseText
```

Sample Code (cont)

```
}        }  XMLHttpRequestObject.send(null);
      }
    }

</script>
 </head>

 <body>

   <H1>Fetching data with Ajax</H1>

   <form>
     <input type = "button" value = "Display Message"
      on click = "getData('data.txt', 'targetDiv')" >
   </form>

   <div id="targetDiv">
    <p>The fetched data will go here.</p>
   </div>

 </body>
</html>
```
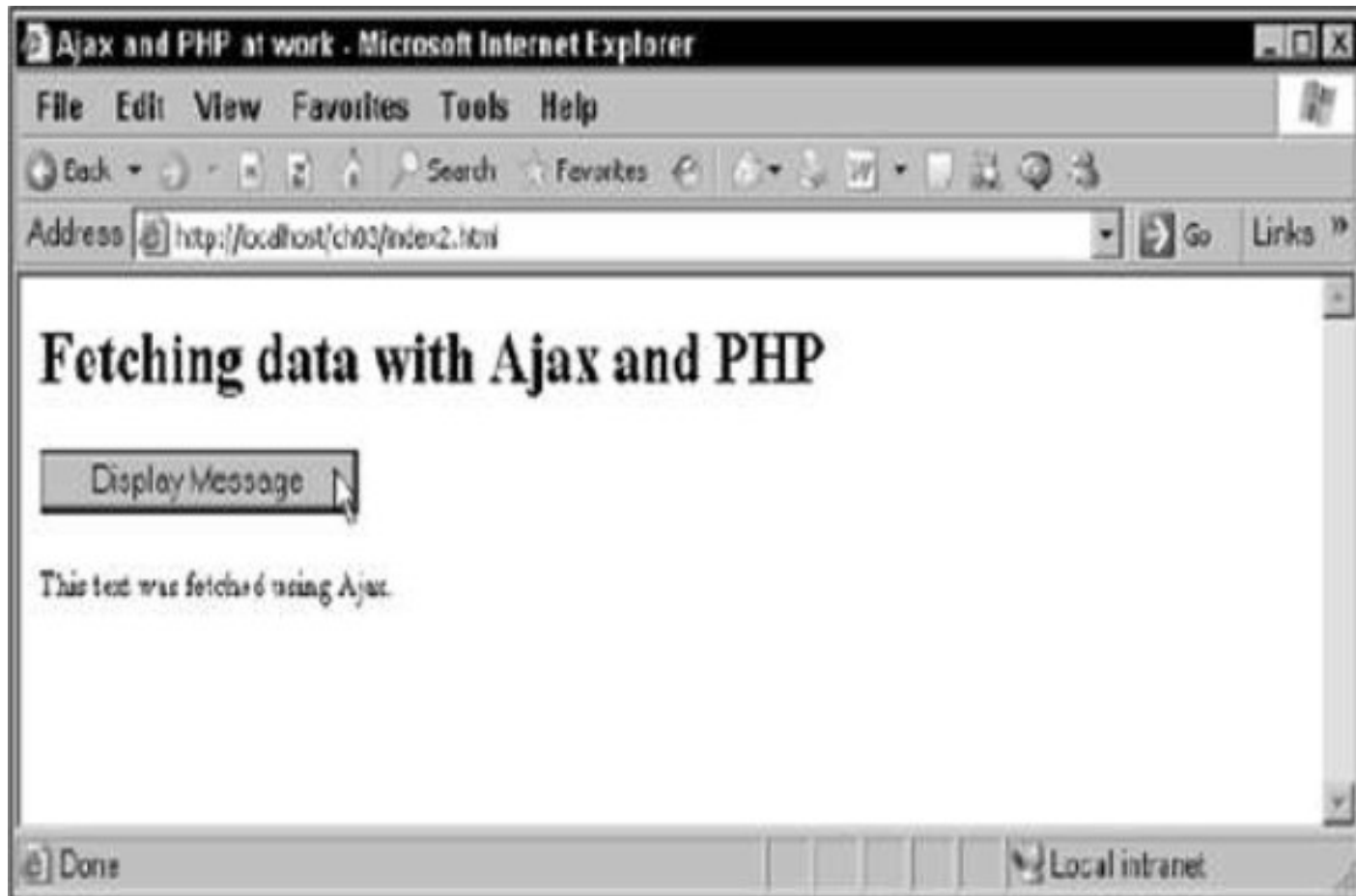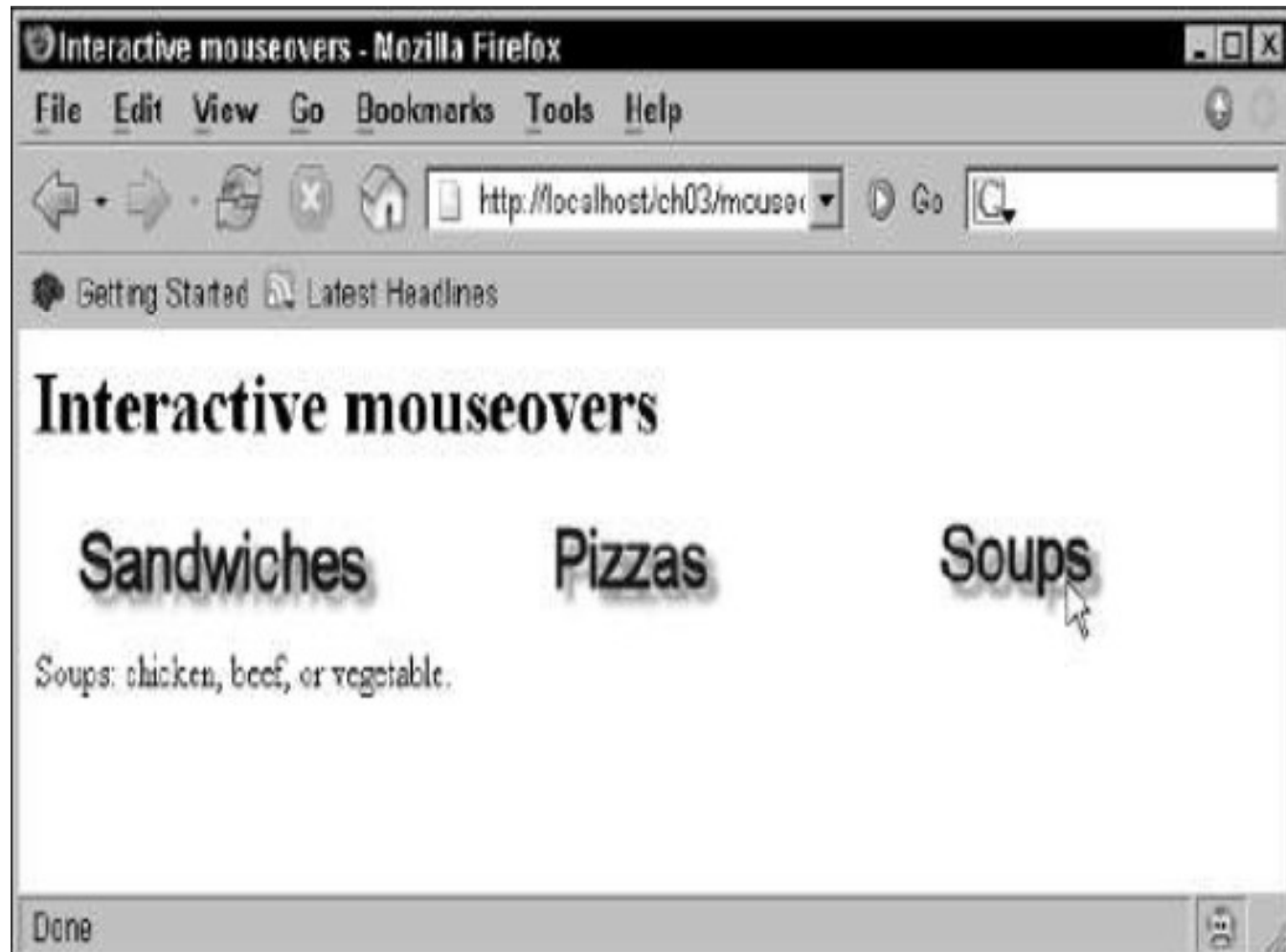
Example

Interactive mouse-overs

Example

- Here comes another Ajax example — one that's a little more impressive visually.

- When you move the mouse over one of the images on this page, the application fetches text for that mouseover by using Ajax.

- All you really have to do is to connect the getData function (which fetches text data and displays it in the <div> element whose name you pass) to the 'onmouseover' event of each of the images you see in Figure

Example

How to do this ….

```
<body>

  <H1>Interactive mouseovers</H1>

  <img src="Image1.jpg"
    onmouseover="getData('sandwiches.txt',
    'targetDiv')">

  <img src="Image2.jpg"
    onmouseover="getData('pizzas.txt',
    'targetDiv')">

  <img src="Image3.jpg"
    onmouseover="getData('soups.txt',
    'targetDiv')">

  <div id='targetDiv'>
    <p>Welcome to my restaurant!</p>
  </div>

</body>
```
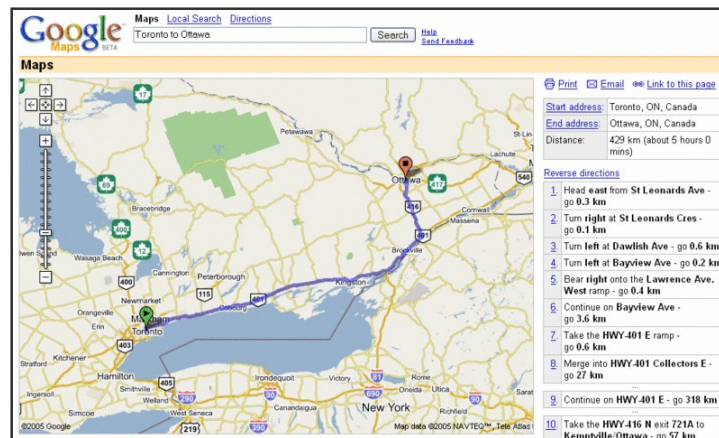
How to do this …

Here's the content of sandwiches.txt :

We offer too many sandwiches to list!

and pizzas.txt :

Toppings: pepperoni, sausage, black olives.

and soups.txt :

Soups: chicken, beef, or vegetable.

Live Examples of AJAX

- [Google](#) suggest is a simple google application which uses Ajax.

References

- http://en.wikipedia.org/wiki/AJAX
- http://www.w3schools.com/ajax/default.asp
- http://www.adaptivepath.com/ideas/essays/archives/000385.php
- http://www.w3.org/TR/XMLHttpRequest/
- http://developer.mozilla.org/en/docs/AJAX